

2019 年全国大学生信息安全竞赛

作品报告

作品名称：基于神经网络及数字水印的图片攻击类别分析系统

电子邮箱：974384602@qq. com

提交日期：2019 年 5 月 31 日

填写说明

1. 所有参赛项目必须为一个基本完整的设计。作品报告书旨在能够清晰准确地阐述（或图示）该参赛队的参赛项目（或方案）。
2. 作品报告采用 A4 纸撰写。除标题外，所有内容必需为宋体、小四号字、1.5 倍行距。
3. 作品报告中各项目说明文字部分仅供参考，作品报告书撰写完毕后，请删除所有说明文字。（本页不删除）
4. 作品报告模板里已经列的内容仅供参考，作者可以在此基础上增加内容或对文档结构进行微调。
5. 为保证网评的公平、公正，作品报告中应避免出现作者所在学校、院系和指导教师等泄露身份的信息。

目录

摘要	4
第一章 作品概述	5
1.1 背景分析	5
1.2 相关工作	6
1.2.1 数字水印	6
1.2.2 机器学习	6
1.3 应用前景分析	7
第二章 作品设计与实现	8
2.1 作品实现概述	8
2.1.1 嵌入过程	8
2.1.2 提取识别过程	8
2.2 数字水印概述	9
2.3 图像数字水印频域算法	9
2.4 水印信息预处理	10
2.5 水印嵌入与提取算法	12
2.6 卷积神经网络	13
第三章 作品测试与分析	16
3.1 测试环境搭建与测试设备	16
3.2 水印的隐蔽性评估	16
3.3 基于 Inception-v3 模型的迁移学习	19
3.4 水印在图像受到攻击后变形的评估	24
第四章 创新性说明	27
第五章 总结	28
参考文献	29

摘要

当今时代是一个信息爆炸的时代，随着网络 and 多媒体技术的发展，多媒体数据逐渐成为人们获取信息的重要来源，但问题也随之而来。由于多媒体信息与传统物质信息相比改动相对简单并且隐蔽性高，有一些不法分子对某些重要的数字信息做出恶意改动，制造出虚假的信息并传播，如果信息的接收人没有辨别出信息的真实性，可能就会造成不可估量的损失。因此，切实有效地保护数字信息的真实性是一个时下非常重要的话题。时下对信息真实性的保护大多在于保护端对端通讯，缺少对大量传播的数据真实性的保护。为了解决这个问题，利用图片受到攻击后水印改动的情况与机器学习的分类算法，我们开发出了基于神经网络及数字水印的图片攻击类别分析系统。该系统能够对插入水印的图片信息做是否受到攻击的判定，如果图像受到改动，还可以对改动的种类进行识别。

本产品由水印嵌入与提取模块、机器学习分类算法模块两部分组成。与其他产品相比，该产品中水印嵌入与提取的算法优不需要原图就可以实现水印的提取，更加切合图像改动检测的实际情况，满足公众对信息真实性验证的需求。应用机器学习对提取后的水印进行分析归类，可以鉴定图像的改动情况并且达到非常高的准确率。在此基础上，还可以扩展其他的功能，比如针对机器学习数据集攻击的识别，在未来的信息时代能够帮助安全从业人员节省大量精力。

关键词：数字水印 攻击检测 机器学习 神经网络

第一章 作品概述

1.1 背景分析

我们生活在一个信息高度发达的时代，手机、ipad、笔记本电脑等现代电子产品为我们每天的生活带来形形色色的信息，传输技术的进步为我们平时生活信息的传递提供了极大的便利，使得信息交流更加迅速准确。然而，信息技术的快速发展也给信息的准确传递带来了挑战。在一个机构公布某些重要信息之后，难免会有一些不法分子别有用心，通过技术手段恶意篡改、攻击一些特殊意义的信息。如果相关人员没有识别出伪造的信息，选择信任了这些虚假的信息，就会造成难以估量的损失。就在 2018 年，惠山法院对全省首例以篡改百度推广信息为手段骗取用户点击的案件进行了宣判。被告人自 2016 年下半年以来，通过自制软件发送钓鱼链接获取百度推广商户的账号和密码，登录这些账号，在原来设置的搜索关键词中添加违法的信息。据介绍，被“坑”的企业遍布北京、上海、江苏、湖南、广东、河南等多个省、市。仅 2017 年 3-5 月间，就造成相关用户经济损失上万元。可见，对公开信息真实性的保护在当下已经凸显出了其重要性。

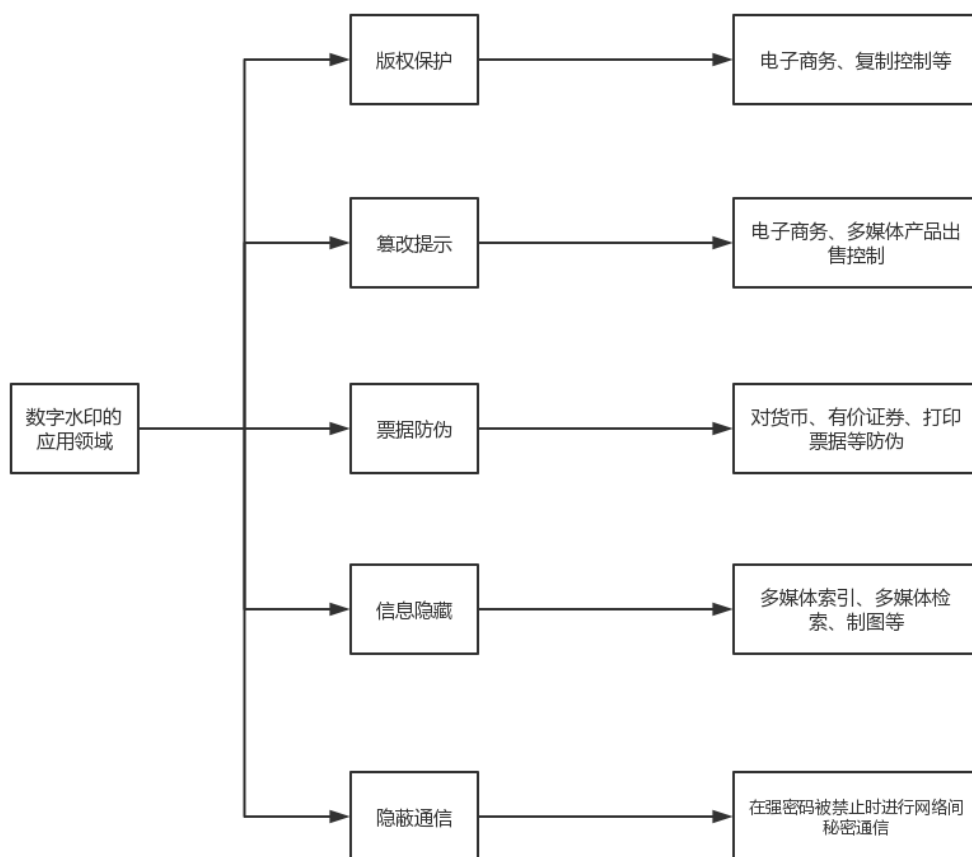
密码学手段也是一种保护信息的方法，但密码学更适合端对端通讯，不适合作为一种面向大众的公开信息真实性的验证方式。对于面向大众公开的信息，作为每一个需要验证信息真实性的用户来说，都生成一组密钥来实现信息的保护是极不方便而且不切实际的。而数字水印作为信息发布者认证信息的一种方式，提取数字水印更加适合让公众去检验信息来源的可靠性。

假设信息发布机构将水印嵌入在面向公众的信息中，公开的信息在人们拿到之后，如果需要对信息的真实性验证，就可以使用本产品的水印提取验证功能，了解信息是否真实是原机构发布的。根据提取水印的形变情况，消息的可靠性就可以辨识。获取到真实有效的信息，才能切实为生活带来便利，提高信息传播的水平。

1.2 相关工作

1.2.1 数字水印

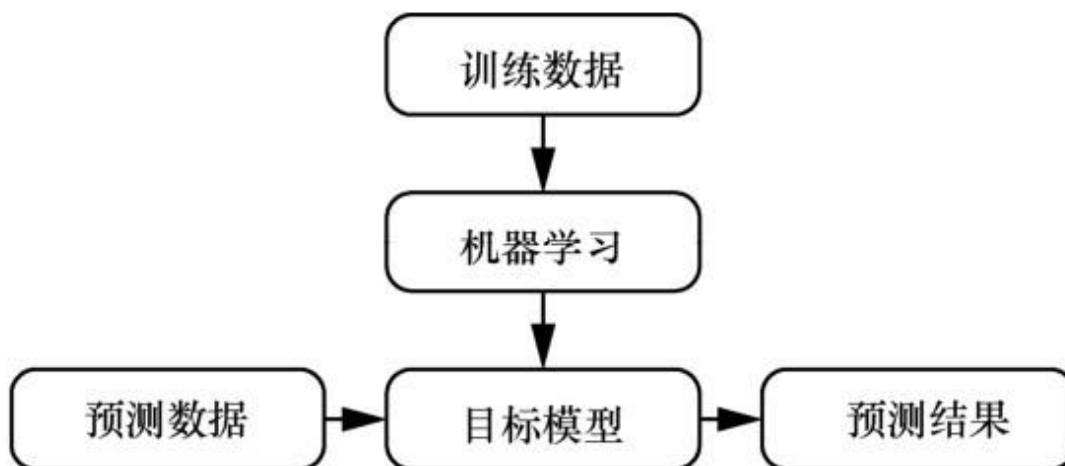
数字水印简单来说就是将信息隐藏到多媒体数字对象中的技术。目前数字水印技术被认为就是利用数学计算方法把具有可鉴别性的数字信息嵌入其他信息中的技术。对于数字水印的应用，具体见下图：



本作品基于数字水印在载体图像受到改动之后的形变,对数字信息做出篡改提示。由于脆弱数字水印在载体图像受到不同种类的攻击时会产生不同的形变,我们可以根据水印的形变状况来判断图像受到的攻击种类。

1.2.2 机器学习

机器学习通过计算手段利用经验改善系统的自身性能。经验即数据,计算机系统利用现有数据进行学习,产生模型进而对未来的行为做出决策判断。



基于本产品的水印算法，对于任意载体图像，只要嵌入的水印相同，其受到攻击后的提取特征也相同。因此，现实中一个公开信息的发布机构只需对应一个水印，本产品可以将需要服务的每个机构对应的水印制作出一组专用数据集进行训练产生模型，模型建立完毕之后即可按照水印提取之后的结果对载体图像进行攻击分类。

1.3 应用前景分析

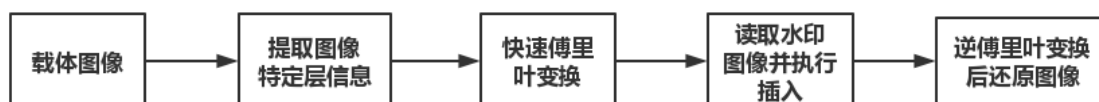
计算机的图像识别技术在公共安全、生物、工业、农业、交通、医疗等很多领域都有应用。例如交通方面的车牌识别系统；公共安全方面的人脸识别技术、指纹识别技术；农业方面的种子识别技术、食品品质检测技术；医学方面的心电图识别技术等。本作品结合计算机图像识别技术，可应用于不同行业信息发布与认证过程中的**真伪检验**。由于水印在载体图像经过不同的改动后会展现出不同的特征，因此通过图像识别来判断图像是否受过改动以及受过何种改动是可行的。同时，应用机器学习来做图像分类可以大大减少相关从业人员的工作量。对于缩放等不改变原图内容的图像变换，鉴定者可根据自身需要进行分别。

第二章 作品设计与实现

2.1 作品实现概述

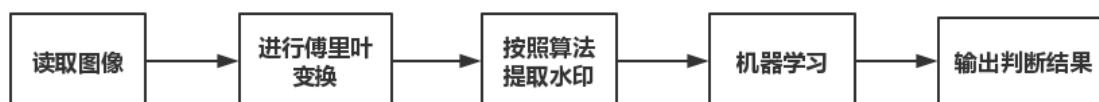
整个作品的实现流程可以分为中央机构使用的嵌入阶段和客户端使用的提取识别过程阶段。具体实现过程如下。

2.1.1 嵌入过程



读取载体图像后提取其中需要操作的数据层,对该数据层进行快速傅立叶变换,取结果后进行水印图像插入,再通过快速傅立叶逆变换实现还原后的图像。

2.1.2 提取识别过程



客户端输入需要识别的图像,该系统读取图像后进行快速傅立叶变换,按照特定算法进行水印提取过程,再通过已经预训练好的模型进行验证分类,输出用户图像相应的信息,判断是否被攻击过,如有则判断受过何种类型的攻击。

2.2 数字水印概述

水印信息为 W ，可以是任何形式的数据；原始载体图片为 I ，密钥 K 作为加强安全性的手段，防止水印受到未授权的攻击，可以是一个数据作为密钥也可即使几个数据的组合；嵌入算法 G 必须具有唯一性，通过密钥 K 唯一的生成添加了水印的图片。水印嵌入算法目前主要分为空域算法和变换域算法。根据上图的基本框架，可以用下面的公式概括水印嵌入算法。

$$W' = G(W, I, K)$$

在提取水印的时候，如果使用原始载体图片 I 或数字水印信息 W 那属于明文水印，如果不使用原始数据则属于数字盲水印，提取算法 G' 也是唯一的，使用与嵌入者相同的密钥 K 则可以提取出来水印，经过这样的处理一般提取出来的水印都会产生信息冗余，难以得到完全的水印还原。用公式概括水印提取算法：

(1) 使用原始载体图片数据 I $W' = G'(I, I', K)$

(2) 使用数字水印信息 W $W' = G'(W, I', K)$

(3) 不使用原始数据 $W' = G'(I', K)$

考虑到本产品的实际应用环境，公众在需要验证信息真伪时是没有原图像用来做提取的。因此有必要实现不使用原始数据做到水印的提取，可以用公式（3）来概括本产品的水印算法。

2.3 图像数字水印频域算法

离散傅里叶变换，简称 DFT，是三种频域变换中最为基础的一种可逆变换，属于整体变换。通过 Fourier 变换就可以在空域和频域中同时处理问题。DFT 水印时利用图像的相位或者幅值进行水印嵌入，而幅值具有 RST 即旋转、尺度、平移操作的不变性，所以在幅值中添加水印信息是主要的方法，得到的水印图片具有抗 RST 的能力。考虑到基于 DFT 算法进行改进后的 FFT 算法具有较高的运算效率，本作品选择 FFT 作为频域变换算法。

2.4 水印信息预处理

本作品的实际应用场景中，中心机构的水印原图是公开的，即客户端和攻击者均可获得中心机构嵌入的水印信息。在作品设计过程中，将置乱矩阵和置乱次数封装进入黑盒，即使攻击者获得了水印原图，由于不知道置乱过程中的具体参数，无法进行伪造水印信息的嵌入，防止没有权限的攻击者伪造中心机构的身份进行模拟签名。现在主要的置乱技术就是用来对水印信息做预处理的。

所谓置乱，就像是拼图一样，把整幅图按像素点划分为，然后整个全部打乱，但总的像素点的个数又不改变。通过这种手段就像是对水印图像进行了加密然后再嵌入，可以很大限度的保证原始水印信息的安全。而这种置乱算法就必须是一种可逆变换，要能够置乱，也能够恢复，才能达到保密的效果。置乱算法有许多种，如果破解者不知道所用的置乱算法，就难以得到原始图像。

置乱算法的总体特点是：

(1) 置乱算法一般都有周期的，随着置乱次数的增加，图片会越来越杂乱、难以辨认，而且通常置乱 1、2 次是没有意义的，等到置乱次数达到周期数就又会回到原始图像。

(2) 置乱后，图像的大小并不会发生改变。不仅可以保证信息隐藏的隐蔽性；而且可以保证水印信息较为均匀的分布到原始图像的各个位置，也使得嵌入水印后的图像对于某些操作，特别是对于剪切、加噪声之类操作的鲁棒性较好。

常用的置换方法有：幻方变换、Arnold 变换、Hilbert 变换、K 采样变换等。此外。还有很多水印预处理方法，如混沌加密。

其中，Arnold 变换具有良好的周期性、编码与解码的特点，在图像传输中可以随机控制变换的次数。利用 Arnold 变换的特性，在图像的置乱处理中已取得有良好的效果，而且在数字水印方面也得到了很好的应用，由于这些特点，本作品决定选用该方法作为水印预处理方法。

具体实现如下：

设数字图像的像素坐标为 $x, y \in \{0, 1, 2, \dots, N-1\}$ ，于是 Arnold 变换为

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \pmod{N}$$

记要变换的矩阵为 A ，右端 $(x, y)^T$ 为输入，左端 $(x', y')^T$ 为输出，考虑到反馈，

迭代程序如下：

$$P_{x,y}^{n+1} = AP_{x,y}^n(\text{mod}N), \quad P_{x,y}^n = (x, y)^T$$

($n=0, 1, 2, \dots$)

式中 n 表示置乱次数， N 为图像的宽和高，也就是说这个置乱只能处理 $N \times N$ 的方阵，其置乱具有周期性。相关研究表明，对于任意 $N > 2$ ，其变换的周期 $T \leq N^2 / 2$ 。

本作品的演示中选择置乱矩阵为 $[[1, 1], [1, 2]]$ ，置乱次数 $n=10$ ，取 $a=b=1$ 进行置乱，结果如下：



置乱之前的图像



置乱之后的图像

可以看出，经过 Arnold 置乱之后的图像已经完全无法辨别。**置乱次数、 a 、 b 参数作为信息发布者的专属消息封入黑盒，必须三个参数全部正确才能认证为消息来源是公开信息发布者。对于需要伪造信息的人来说，同时攻破 3 个参数是很难的。**

还原过程中使用置乱矩阵的逆矩阵以及相同的置乱次数 n 和 a, b 取值，可以将提取出的图像进行置乱还原得到原水印。

妄图伪造中心机构的攻击者在获得需要嵌入的水印的前提下，可以进行水印嵌入。但正式由于置乱机制的存在，该攻击者无法获得置乱矩阵以及 a 、 b 、 n 的取值情况，嵌入的水印图像与中心机构置乱处理之后的图像存在很大区别。这样，使用中心机构提供的检验工具进行检验提取时，提取出的水印信息与原嵌入水印差别很大，无法通过原图正常的检测验证过程。**通过该方法可以防止攻击者伪造中心机构进行水印添加，**

从而发布不具有官方效应的假信息。

2.5 水印嵌入与提取算法

对于一张需要嵌入的水印彩色图像，由于彩色图像的信息是三维数组，而我们使用的算法是二维离散傅里叶变换，因此选择要嵌入水印图像的 RGB 通道的任意一层进行水印的嵌入。读取一层的图像信息，进行一次二维离散傅里叶变换。由于在高频区域嵌入水印的隐蔽性更好，因此本作品选择把水印的信息嵌入在高频位置。

选取一张作为信息发布机构标志的二值图像作为水印图像，我们可以选取黑色像素点或白色像素点中的任意一种的位置作为嵌入的信息。具体嵌入方法如下：

- 1、 读取想要在载体图像中嵌入水印的 RGB 通道中的一层，本作品读取 G 通道数据进行操作；

- 2、 对该层数据进行二维快速傅里叶变换；

- 3、 读取二值水印，在二值水印对应的矩阵上记录黑色像素块或白色像素块的位置信息。

- 4、 选取图像频域图中高频区域的一块位置，将该位置与二值水印对应的矩阵创立映射，将作为代表黑色或白色的像素点的数字信息嵌入到对应的高频区域中，具体嵌入算法为乘性相加。本作品将黑色像素点对应的位置识别为 1，即在该位置数据的实部上进行乘性相加，如果取 $N=10000$ ，则加入的数据为 $a*N+b$ (即 10001)。

- 5、 以中心点为对称点，在与这块区域相对称的位置以同样的方式嵌入信息使傅里叶变换后的矩阵数值依然为中心对称。

- 6、 对加入水印信息的频域矩阵做二维快速傅里叶逆变换，就生成了嵌入水印信息之后的图像。(嵌入后图像的相似度对比即量化衡量在下文提及)

水印提取的过程为上述过程的逆过程，具体实现如下：

- 1、 选取需要提取图像对应的 RGB 通道并进行数据分离。

- 2、 对嵌入水印的图像做二维快速傅里叶变换。

- 3、 找到嵌入的高频位置块。

- 4、 根据嵌入时选择的黑色像素点与白色像素点对应的嵌入信息做数值相关性检测，按照一定的阈值进行数据提取，将提取的二值数据绘制进入新的图像中，从而实现嵌入水印的恢复过程。

2.6 卷积神经网络

本作品的机器学习部分采用卷积神经网络（Convolutional Neural Networks, CNN）实现。

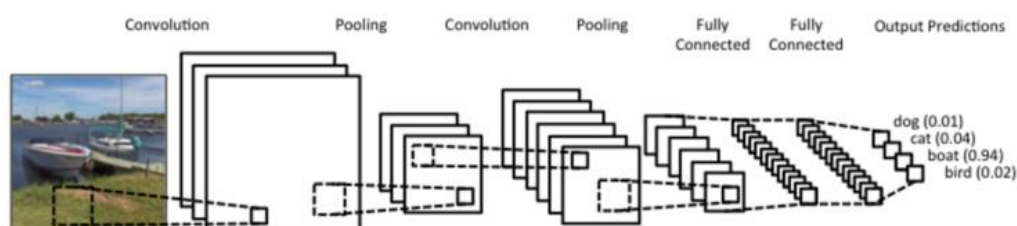
卷积神经网络（Convolutional Neural Network）简称 CNN，CNN 是所有深度学习课程、书籍必教的模型，CNN 在影像识别方面的为例特别强大，许多影像识别的模型也都是以 CNN 的架构为基础去做延伸。

卷积神经网络是一种多层的监督学习神经网络，隐含层的卷积层和池采样层是实现卷积神经网络特征提取功能的核心模块。该网络模型通过采用梯度下降法最小化损失函数对网络中的权重参数逐层反向调节，通过频繁的迭代训练提高网络的精度。卷积神经网络的低隐层是由卷积层和最大池采样层交替组成，高层是全连接层对应传统多层感知器的隐含层和逻辑回归分类器。第一个全连接层的输入是由卷积层和子采样层进行特征提取得到的特征图像。最后一层输出层是一个分类器，可以采用逻辑回归，Softmax 回归甚至是支持向量机对输入图像进行分类。

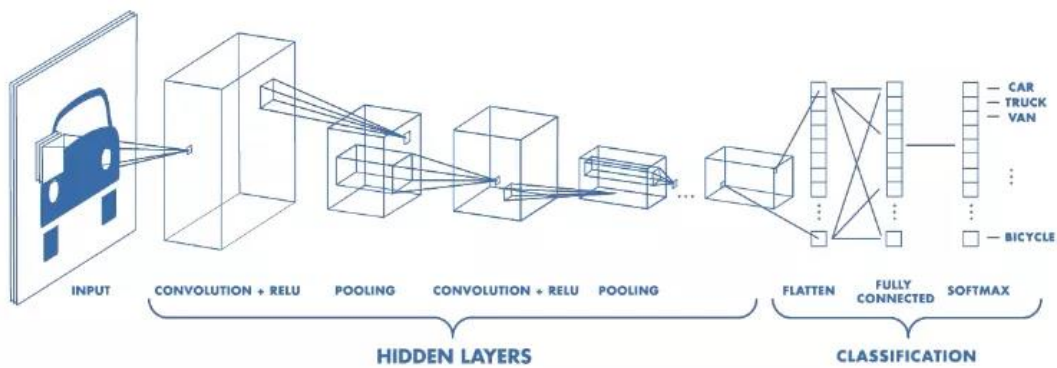
卷积神经网络结构包括：卷积层，降采样层，全链接层。每一层有多个特征图，每个特征图通过一种卷积滤波器提取输入的一种特征，每个特征图有多个神经元。

输入图像统计和滤波器进行卷积之后，提取该局部特征，一旦该局部特征被提取出来之后，它与其他特征的位置关系也随之确定下来了，每个神经元的输入和前一层的局部感受野相连，每个特征提取层都紧跟一个用来求局部平均与二次提取的计算层，也叫特征映射层，网络的每个计算层由多个特征映射平面组成，平面上所有的神经元的权重相等。

通常将输入层到隐藏层的映射称为一个特征映射，也就是通过卷积层得到特征提取层，经过 pooling 之后得到特征映射层。

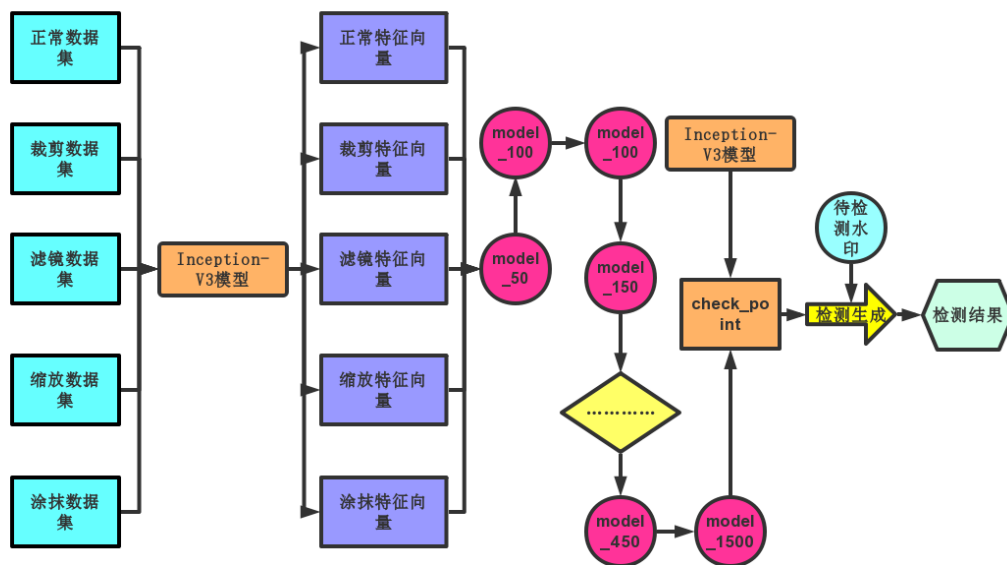


CNN 概念图



CNN 概念图

由于其良好的图像识别方面的性能，我们最终选定 CNN 作为我们对图像进行攻击分类的实现方法。



我们最终采用了如上图的整体识别方案，整体流程大概可以分为三部分：

- 数据预处理
- 模型构建
- 模型训练

数据预处理主要解决的问题是样本与攻击类型的匹配。首先进行的样本制作，对植入过水印的图像进行分类别攻击，生成大量经过攻击的图像，接下来的水印提取和水印筛选，除去一些在格式转换过程中产生的错误水印样本避免对数据集的污染，符合条件的水印为模型训练提供了标准规范化输入。

模型构建主要解决的问题是构建一个高效的模型来实现数据特征提取和攻击分类；提取出的水印难以通过常规方法刻画特征进行分类，因而结合卷积神经网络的优秀的图像特征刻画能力提出了使用卷积神经网络提取特征的思想，然后考虑到数据集的数量较少和计算资源有限的事实，结合迁移学习最终构建出一个基于迁移学习的 CNN 网络。

最后经过不断地参数调整训练出了一个分类正确率在 94%左右的模型，并且通过实验证明了训练出的模型具有很强的鲁棒性。

第三章 作品测试与分析

3.1 测试环境搭建与测试设备

CPU: corei7-7700hq

GPU: GTX 1060 6G

RAM: 8G

Tensorflow 1.18

python 3.6 以上版本

matlab 2018

Windows 10 64 位系统

3.2 水印的隐蔽性评估

下面通过进行比较嵌入水印前后的照片来对水印的隐蔽性进行评估。



嵌入水印之前



嵌入水印之后

我们的作品一个很重要的特性就是隐蔽性。因为是以各大信息发布平台为潜在应用场景，所以务必要使水印嵌入对图像的影响达到最低。

使用峰值信噪比(Peak Signal-to-Noise Ratio, PSNR)来估计水印嵌入宿主图像之后含水印图像的扭曲程度，并作为反映一个数字水印算法不可见性好坏的指标。尽管这个指标与肉眼观察有出入，但是可以作为一个比较好的经验规则来衡量水印的隐蔽性。我们通常使用 PSNR 作为衡量标准，对于常见的图像格式，灰度级为 0-255。0 表示的是黑颜色，255 表示的是白颜色。PSNR 的求解公式如下所示：

$$PSNR = 10 \log_{10} \frac{255^2}{MSE}$$

在峰值信噪比计算过程中，需要考虑均方误差 MSE（均方误差），所对应的表达式如下所示

$$MSE = \frac{1}{M \times N} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} (f(x, y) - f_{\tau}(x, y))^2$$

这里所表示的是相对应的原始图像中包含的像素值， $f(x, y)$ 表示的是对应的添加

水印以后所得出的图像对应的像素值，M，N 表示图像的宽、高。

PSNR 值的范围满足条件为 20~40 分贝之间，该值的值一般选择精确到保留小数点后两位，如 42.03。PSNR 是通过两幅或多幅图像计算而得到的值，可以很好的反映图像质量。

隐蔽性评估我们选择了在 matlab 中进行仿真，评估所用代码如下：

```
>> img=imread('嵌入之前.png');
[h w]=size(img);
imgn=imread('嵌入之后.png');
img=double(img);
imgn=double(imgn);

B=8;                %编码一个像素用多少二进制位
MAX=2^B-1;          %图像有多少灰度级
MES=sum(sum((img-imgn).^2))/(h*w);    %均方差
PSNR=20*log10(MAX/sqrt(MES));         %峰值信噪比
PSNR
```

输出：

```
PSNR(:, :, 1) =

    44.6222

|
PSNR(:, :, 2) =

    Inf

PSNR(:, :, 3) =

    Inf
```

如果 PSNR 低于 20db，说明前后图像相差过多；PSNR 在 20 到 40db 之间，表示图像质量尚可；PSNR 高于 40dB，说明嵌入前后图像相似度相当高（即非常接近原始图像）。

我们的 PSNR 为 44.62，表明嵌入之后图像几乎没有变化，嵌入水印对图像的影响近乎于无，隐蔽性非常好。

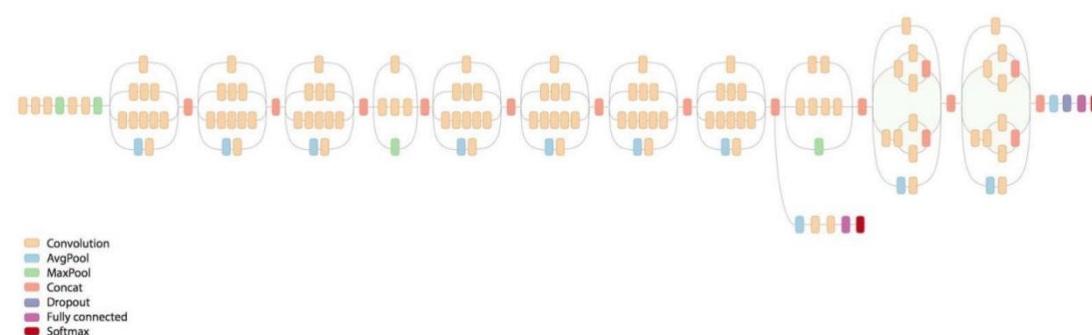
3.3 基于 Inception-v3 模型的迁移学习

考虑到本作品目前难以收集到大量数据集进行模型训练，所以需要找出一种在数据量较小的情况下保证模型的识别的方法。

目前业界针对小数据集进行卷积神经网络训练提出的主流方式是迁移学习（transfer learning）。该方法的思想是使用大数据集进行模型训练，然后使用训练好的参数对小样本进行参数微调，这样使得最终训练出来的模型具有小数据集和高精度的双重优势，由此可见本作品数据集上理想的解决方案即是：迁移学习+参数微调。

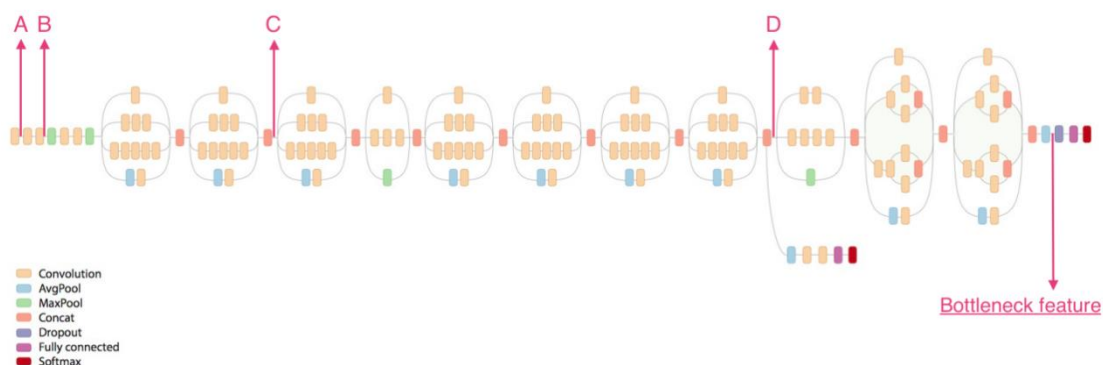
迁移学习目前网络上提供的模型较多，综合考虑模型参数量、模型表现、模型可得性、模型训练计算资源需求量、模型易用性等因素，本作品选择了 Google 开源的 Inception-v3 模型作为迁移学习的框架，该框架在 ImageNet 数据集上进行训练，该数据集有着 1000 个类别超过 100 万张图片数据，该模型在 ImageNet 数据集上获得了优秀的表现。

Inception-v3 的网络结构如下图：



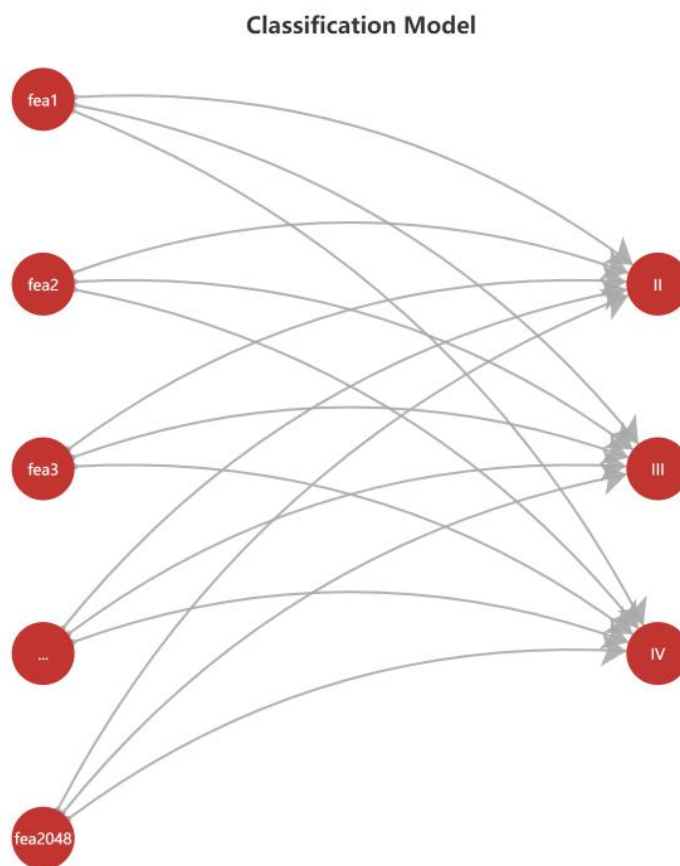
选择了 Inception-v3 模型后，将该模型瓶颈层的输出作为提取的特征，然后把

这些特征经过一个全连接层进行分类。



在上图中最右边示意的 Bottleneck feature 位置即是特征输出的位置，也就是说该模型的最后三层被丢弃，然后瓶颈层的结果作为新模型的特征提取结果，该结果是 2048 长度的特征向量。

获得该特征向量之后，需要将其输入一个全连接层进行分类，全连接层的模型示意图如下：

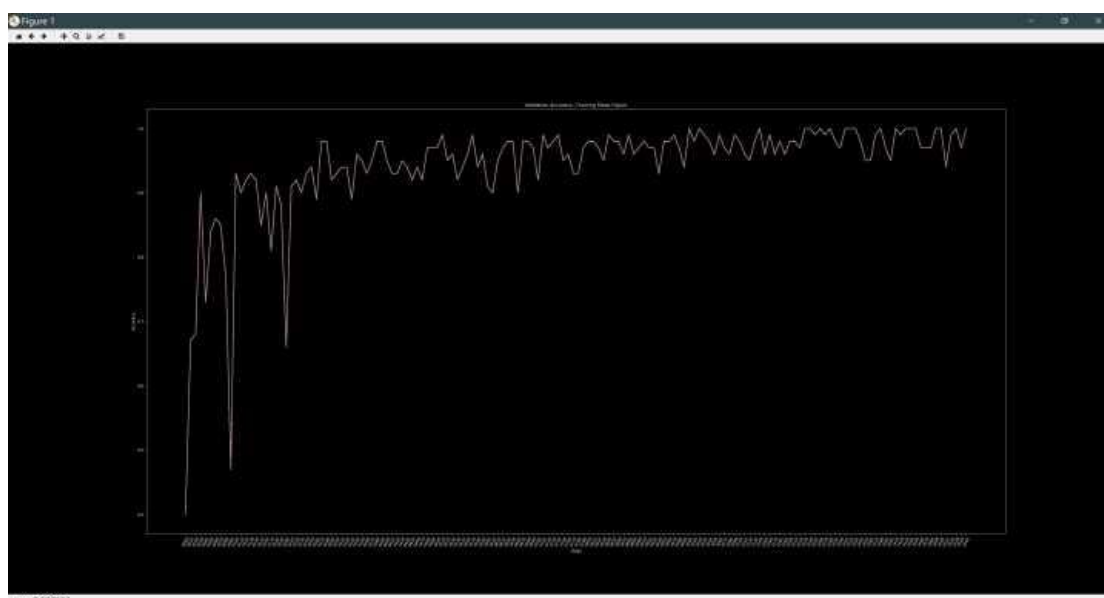


将以上两个模型连接到一起就构成了完整的模型，模型的输入是目标图片提取出的数字水印，而输出就是对图片攻击种类（包括正常）判断。

训练过程中的参数如下：

- 学习率：指数衰减学习率，初始 0.1，衰减极限 0.01；
- 每次输入的数据量：100
- 测试集占比：10%
- 验证集占比：10%

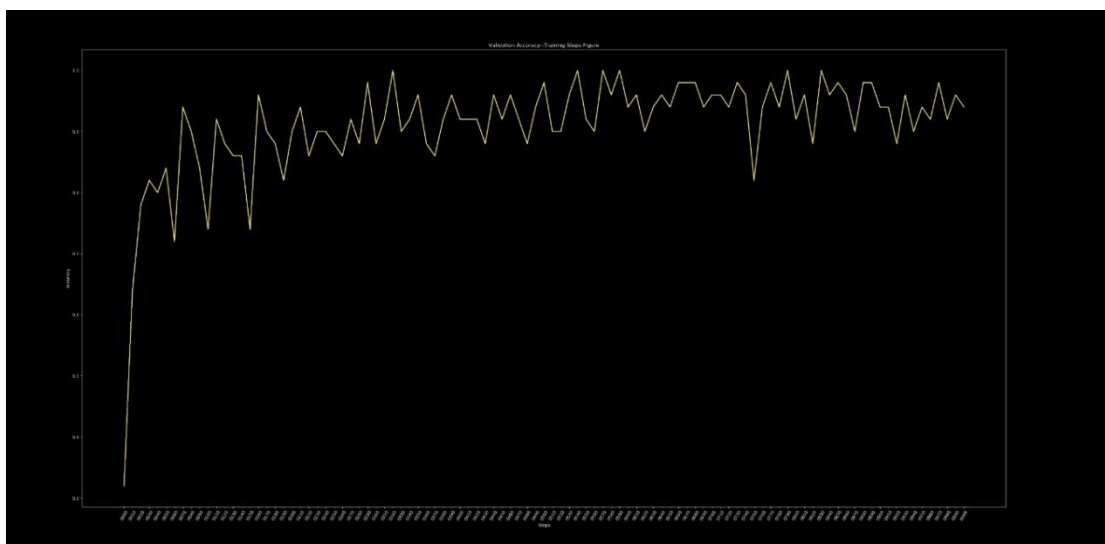
在测试过程中训练效果表现较好，其中一次训练步数为 1000 次训练效果如下：



```
C:\Windows\system32\cmd.exe - python G:\Inception-v3\Classifier.py
Saved model checkpoint to C:\Users\LGJhr\Desktop\作品赛产出\runs\1558971832\checkpoints\model-1350
Step 1360 :Validation accuracy on random sampled 100 examples = 95.0%
Step 1370 :Validation accuracy on random sampled 100 examples = 99.0%
Step 1380 :Validation accuracy on random sampled 100 examples = 100.0%
Step 1390 :Validation accuracy on random sampled 100 examples = 97.0%
Step 1400 :Validation accuracy on random sampled 100 examples = 95.0%
Saved model checkpoint to C:\Users\LGJhr\Desktop\作品赛产出\runs\1558971832\checkpoints\model-1400
Step 1410 :Validation accuracy on random sampled 100 examples = 100.0%
Step 1420 :Validation accuracy on random sampled 100 examples = 99.0%
Step 1430 :Validation accuracy on random sampled 100 examples = 100.0%
Step 1440 :Validation accuracy on random sampled 100 examples = 100.0%
Step 1450 :Validation accuracy on random sampled 100 examples = 100.0%
Saved model checkpoint to C:\Users\LGJhr\Desktop\作品赛产出\runs\1558971832\checkpoints\model-1450
Step 1460 :Validation accuracy on random sampled 100 examples = 97.0%
Step 1470 :Validation accuracy on random sampled 100 examples = 97.0%
Step 1480 :Validation accuracy on random sampled 100 examples = 97.0%
Step 1490 :Validation accuracy on random sampled 100 examples = 100.0%
Step 1500 :Validation accuracy on random sampled 100 examples = 100.0%
Saved model checkpoint to C:\Users\LGJhr\Desktop\作品赛产出\runs\1558971832\checkpoints\model-1500
Step 1510 :Validation accuracy on random sampled 100 examples = 94.0%
Step 1520 :Validation accuracy on random sampled 100 examples = 99.0%
Step 1530 :Validation accuracy on random sampled 100 examples = 100.0%
Step 1540 :Validation accuracy on random sampled 100 examples = 97.0%
Step 1549 :Validation accuracy on random sampled 100 examples = 100.0%
Final test accuracy = 98.4%
```

仅仅一千步，通过少量数据训练的效果就稳定在 98.4%，可以说效果非常良好，这个模型在实际场景中测试的效果也十分良好，对几种常见图片攻击都可做到有效识别。

训练过程中的测试准确率图像大致如下图：



训练产生的模型如下：

checkpoint	2019/5/25 17:56	文件	1 KB
model-750.data-00000-of-00001	2019/5/25 17:55	DATA-00000-OF...	41 KB
model-750.index	2019/5/25 17:55	INDEX 文件	1 KB
model-750.meta	2019/5/25 17:55	META 文件	93,514 KB
model-800.data-00000-of-00001	2019/5/25 17:56	DATA-00000-OF...	41 KB
model-800.index	2019/5/25 17:56	INDEX 文件	1 KB
model-800.meta	2019/5/25 17:56	META 文件	93,514 KB
model-850.data-00000-of-00001	2019/5/25 17:56	DATA-00000-OF...	41 KB
model-850.index	2019/5/25 17:56	INDEX 文件	1 KB
model-850.meta	2019/5/25 17:56	META 文件	93,514 KB
model-900.data-00000-of-00001	2019/5/25 17:56	DATA-00000-OF...	41 KB
model-900.index	2019/5/25 17:56	INDEX 文件	1 KB
model-900.meta	2019/5/25 17:56	META 文件	93,514 KB
model-950.data-00000-of-00001	2019/5/25 17:56	DATA-00000-OF...	41 KB
model-950.index	2019/5/25 17:56	INDEX 文件	1 KB
model-950.meta	2019/5/25 17:56	META 文件	93,514 KB

识别程序结合训练好的模型和 Inception-v3 模型即可进行识别。该模型的数据集制作过程纳入了大量不同种类的图片，对于不同的载体图像，做出上文中提到的几种图像变换方式，提取出的水印都拥有共同的特征，这就对模型的制作带来了极大的便利，同时使用一个模型就可以满足对一个水印的所有检测。

每张图片输入进终端后将自动进行水印提取处理及类型识别结果如下：（处理后的图像与水印变化情况见 3.4 中表格）

```
(tensorflow) λ python check.py
启动中，请稍后...
请输入需检测的图像路径：正常.png

水印提取结束！开始检测，请稍后...
WARNING:tensorflow:From E:\Anaconda3\lib\site-packages\tensorflow\python\ops\gen_ops.py:17:
nd will be removed in a future version of TensorFlow. For more information, see
Instructions for updating:
Use standard file APIs to check for files that exist: tf.gfile.Exists(filename)

检测结果：
图片正常
```

```
请输入需检测的图像路径：caijian.png

水印提取结束！开始检测，请稍后...

检测结果：
图片被裁剪过！
```



```
请输入需检测的图像路径: suofang.png
水印提取结束! 开始检测, 请稍后...

检测结果:
图片被缩放过!
```

```
请输入需检测的图像路径: tumo.png
水印提取结束! 开始检测, 请稍后...

检测结果:
图片被涂抹过!
```

```
(tensorflow) > python check.py
启动中, 请稍后...
请输入需检测的图像路径: lvjing.png

水印提取结束! 开始检测, 请稍后...
WARNING:tensorflow:From E:\Anaconda\envs\
ts (from tensorflow.python.training.che
Instructions for updating:
Use standard file APIs to check for fil

检测结果:
图片添加过滤镜!
```

所测试的图片皆通过了测试证明了模型的准确性。


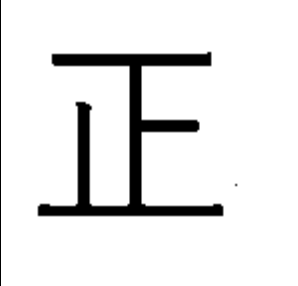


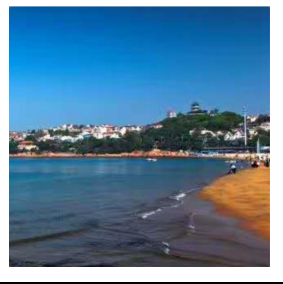
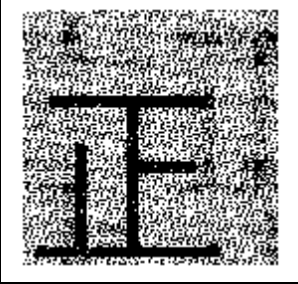
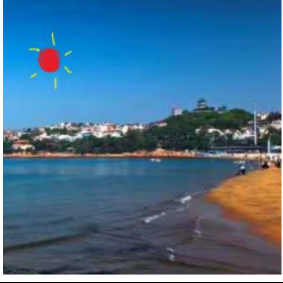
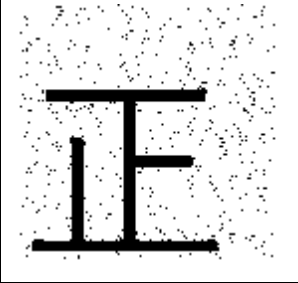
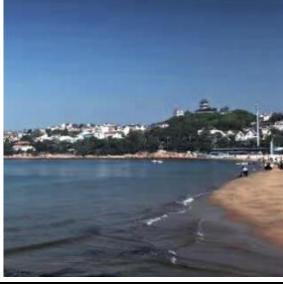
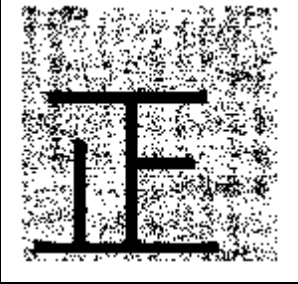
3.4 水印在图像受到攻击后变形的评估

我们产品的第二大特性就是水印对攻击类别的敏感性。对此, 我们将针对各种类型, 各种程度的图像攻击, 比较它们对嵌入水印的图像的影响, 并且用我们已经训练好的迁移学习模型对提取出的水印进行检测, 以此来测试我们的模型的准确性。

以下对图像受到攻击之后水印的变形进行分析, 原水印我们采用下面这张图片:

正

添加的水印

变化类型	变化后的含水印图像	变化图像提取出的水印	检测结果 (程序运行结果在 3.3 中已展示, 此处不再重复截图)
正常			正常
裁剪变化			裁剪
缩小至 95%			缩放
涂抹变化			涂抹
滤镜变化			滤镜

从我们统计的表格中可以看到，即使是肉眼看上去非常细微的变化，所提取出的水印都和原水印有着天壤之别，由此可以看出我们的水印对于攻击的敏感性；而对于同一类型不同程度的攻击，我们对提取出水印进行比较，肉眼难以察觉其中的规律，但是机器能够办到。**经过我们的 V3 模型训练，正常范围内的攻击类型都被我们检测出来了，准确率相当之高。**

如果把我们的产品应用于各大信息发布平台，便可以很容易地在不需要原图的情况下判断出各种来源的图片是否来自该平台，有没有遭受过恶意攻击，具体是哪种类型的攻击。这种特性对于信息验伪来说非常方便。

第四章 创新性说明

1、 基于神经网络迁移学习的盲水印图像攻击检测

本方案开创性地将图像攻击检测与数字水印、神经网络图像识别结合，通过神经网络检测载体图像经过不同变化之后水印的形变情况来判断原图受到哪种类型的改变，可以对像素点集的篡改做出反应，对于原图的敏感性非常强，精度可以达到相当高的水平；同时本作品采用了迁移学习的方案，大大提升了模型训练效率，同时减少了投入的算力成本。

2、 无需原图即可检测图像真实性

大部分水印的提取都需要用到未被改动过的原图像，综合考量本产品的实际应用情况，在实际生活中做改动检测是没有原图像作为基础的，因此必须在没有原图像的情况下提取出一个比较完整的水印以供检测。我们的方案完美实现了这一点，通过算法在没有原图的情况下提取出水印；同时，水印对图像的影响近乎于无，隐蔽性非常好，大大拓宽了本方案潜在的应用背景。结合大量水印数据集的支持，我们的模型可以直接在没有原图的情况下检测水印，精确检测出原图所受过的攻击类型，在实际场景之下更具有使用价值。

3、 实现了精准的图像攻击类型分类

本作品通过自己制作的大量水印数据集结合深层卷积神经网络，在经过训练之后，将图像攻击种类的分类精度提高到 94%以上，而对于单纯的是否遭到攻击的检测则可以达到更高的精度，根据现有数据估测，目前的准确率水平已经高于人工识别的准确度。

4、 对于肉眼不可视攻击也可以做到精确检测，可能成为图像数据集保护的新思路

本作品的检测基于数字盲水印而不依赖于图像本身，对图像数据进行的任何改动都会影响到水印，以此达到对肉眼不可视攻击的检测。机器学习的数据集本身也是攻击者攻击的目标，现在的攻击者往往通过肉眼不可视的攻击方法攻击图像数据集，虽然本水印也会对图像造成一定影响，但水印的影响是可以预知并用相应手段消除的，如果可以通过相关验证，那么本作品在未来也可能成为图像数据集保护的新思路。

第五章 总结

今天，知识成为社会发展的核心驱动力，数字化飞速发展，知识的获取变得越来越容易，消息的来源认证就变得越来越重要。而数字水印在消息来源认证过程中扮演着重要的角色。数字水印提出已经二十余年，许多专家、研究人员对此进行了深入的研究，提出了许多相当成熟的成果。

本作品选择了 DFT 域变换，结合其他比较成熟的算法和思路。从结果来看，选择的数字水印算法各种特性表现良好，本算法的优点是：

- (1) 提取水印不需要原始数据的参与，适合于大多数实际操作情况。
- (2) 对于各种类型的图像篡改提取结果区分特征明显，为与机器学习结合识别图像的改动提供了可能。

本方案致力于解决图像传播过程中的信息验证问题。**工作流程共分为四个部分：水印嵌入、水印提取、模型训练、水印检测。**通过对经过不同变化的嵌入水印的图像的水印提取，我们建立起了大量不同类型对应的数据集。在此基础上通过 Inception-V3 框架进行迁移学习的训练，我们的系统就掌握了绝大部分变化图像所提取出的水印的特征。可以在没有原图像的情况下做到无论图像收到过什么类型，什么程度的改动，我们都能让机器迅速识别出来，实现高效稳定的信息验伪。与此同时，我们的水印算法基于快速傅里叶变换，对于原图的影响非常之小，通过 PSNR 检测，我们可以确定嵌入水印前后的图像峰值信噪比基本大于 40，隐蔽性非常好。水印嵌入与提取不需要原图就可以实现水印的提取，更加切合图像改动检测的实际情况，满足了实际应用场景下公众对信息真实性验证的需求。在此基础上，还可以扩展其他的功能，比如针对机器学习数据集攻击的识别，在未来的信息时代能够帮助安全从业人员节省大量用于人工清洗数据集精力。

参考文献

- [1] 刘永生. 基于变换域数字水印算法的研究 [D]. 江苏: 南京信息工程大学, 2007.
DOI:10.7666/d.y1080186.
- [2] 宋蕾, 马春光, 段广晗. 机器学习安全及隐私保护研究进展 [A]. 网络与信息安全学报, 2018, Vol. 4(8): 16-29.
- [3] 郑林涛, 阎晓东, 范延军, 孙燮华. 一种基于块均值的盲检测水印算法 [A]. 中国计量学院学报 15(4): 0281~0283, 2004
- [4] 董兵, 张建伟. 一种基于相关值检测盲水印方法 [A]. 计算机应用与软件, Vol. 24 No. 4 Apr. 2007